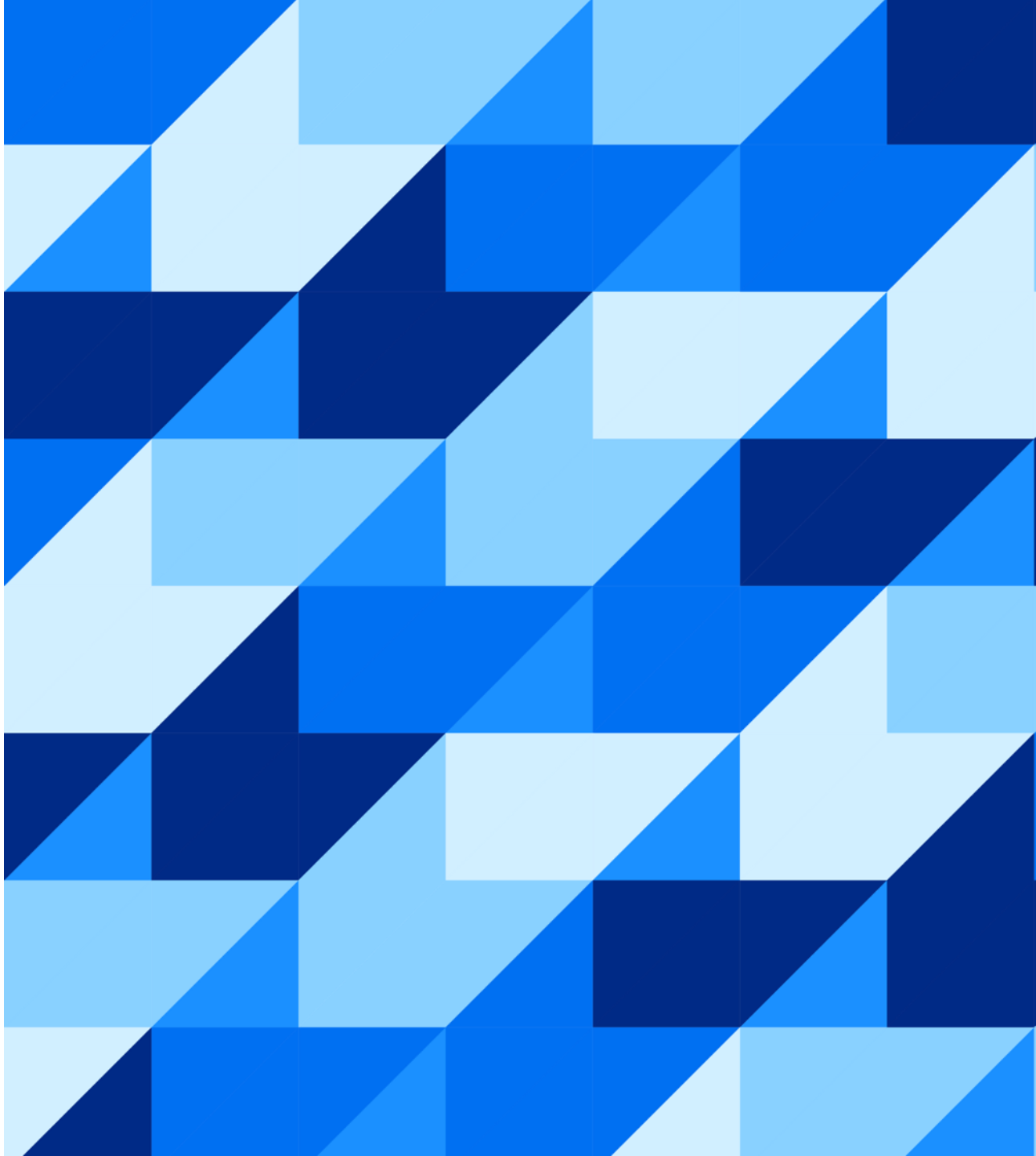


SAP Community Content: Monitoring in SAP Datasphere

Olaf Fischer, SAP

Agenda

- Monitoring Paradigms
- Monitoring Samples
- Standard Monitoring
- General Content Capabilities
- Reporting: Countring / Measure as Dimension
- Reporting: Dimension Overview
- Reporting: Authorizations
- Modelling Session – Q&A

Monitoring Paradigms

Understanding of ”Monitoring” in General

Monitoring is associated with a certain intention

- Check the performance of the system (technical)
- Validation of user adoption for projects (business)
- Usage and activity tracking (who has done what at which time, audit)
- Replication issues (data quality)
- ... and more ...

Monitoring is not made for ...

- ... returning the root cause of an issue
- ... explaining why tasks takes long
- ... proving hints to optimize

Monitoring Matrix

	TOP N	Health	Course of Time
HANA *(Global)*	• Do we see expensive statements in the overall HANA DB? • Which statements can be optimized?	• Do we see any peaks or irregular occurrences? • Are these peaks reoccurring the same time pattern?	Higher consumption in CPU, memory or higher duration is observed
Tasks *(Datasphere)*	Identify the statements with the highest duration, CPU and memory	• Do we see failed tasks? • Does the runtime of the same task has high variances?	• Do we see an increase in average CPU and/or memory consumption? • Impact of a new Datasphere or HANA release.
InA/MDS *(SAC, OData)*	Identify the InA/MDS statements with the highest duration, CPU and memory.	• Are there any InA/MDS statements causing OOM errors? • Is the response time somehow stable and reproducible?	• Are there any InA/MDS statements which increased over time in CPU, memory or duration? • Can we observe less user activity?

Modelling Samples

Monitoring – Introduction

Define the two spaces dedicated to monitoring in SAP Datasphere (such as monitoring the database for resource consumption).

SAP_ADMIN space: This space is dedicated to the pre-configured monitoring views provided as business content by SAP via the Content Network.

Expensive Statement Tracing: switch on.

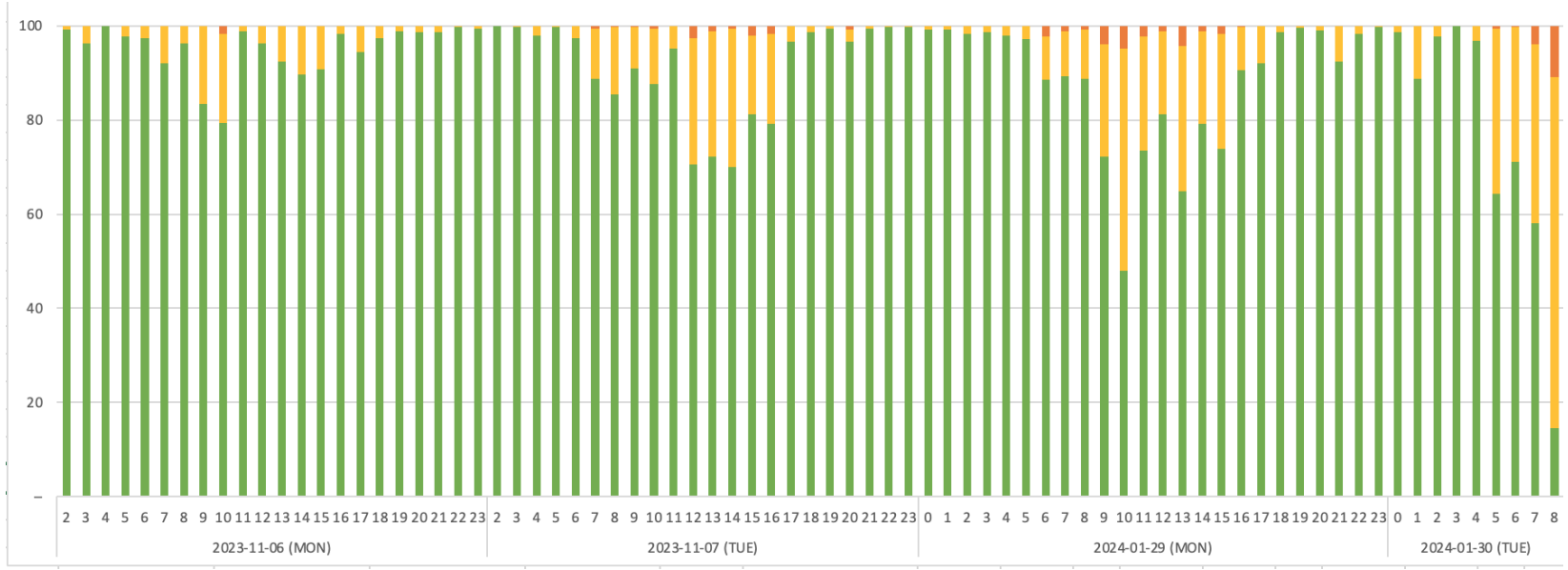
MDS Tracing: switch on.

Monitoring – Health per Hour

Health per hour allows you to see the distribution for CPU and memory on an hourly basis, providing you an overview of the general health of the tenant. Are there many executions consuming more than 80 % of the available capacity?

Analytic Model: TCT_HANA_COCKPIT_AM_01

Exemplary Settings:
Column: Measure(Count), CPU as COL → select %-segment
Row: Date-Day *Filter on Port of Indexserver (typically 30040)*



Exemplary visualization

Monitoring – MDS – reliable response time

Monitor the response time to check if overall response time is relatively constant or shows some peaks during a specific timeframe.

Analytic Model: TCT_MDS_MEDIAN_AM_01

Exemplary Settings:

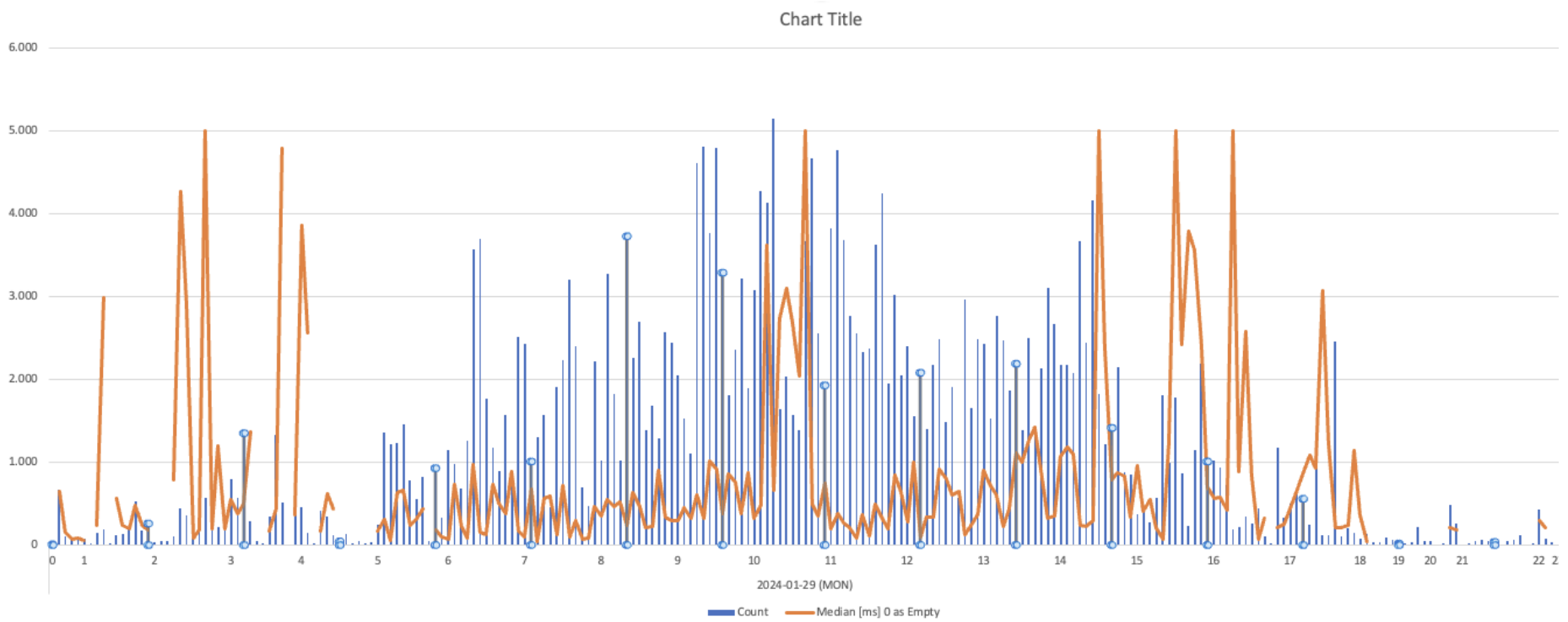
Rows:

- Date-Day
- Hours
- 5-minute

Columns:

- Measure(Median)
- Measure(Count)

Exemplary visualization



Monitoring – Failed Tasks 1/3

Find out which tasks failed the most in the system to troubleshoot and identify causes to avoid negative impact on the system due to e.g. outdated data.

Analytic Model: TCT_TASK_LOGS_AM_01

Exemplary Settings:
Filter on STATUS = FAILED
Rows:

- Date → Date-Day,
- Activity,
- Application ID

Column:

- Measure(Count)

				Measures	ROW_COUNT
START_DATE	Weekday	ACTIVITY	APPLICATION_ID	OBJECT_ID	
2024-02-12	MON	EXECUTE	DATA_FLOWS	DF_ZCACTEMP_NEW	8
				DF_ZCCASESLOADED	18
				DF_ZCCASESPACKED	10
				DF_ZCCASESPICKEDB	16
				DF_ZCDELVCARR	19
				DF_ZCDELVITEMSIB	7
				DF_ZCDELVITEMSOB	14
				DF_ZCDELVPROCOVB	11
				DF_ZCIBDELV	7
				DF_ZCINBFILTER	6
				DF_ZCINBFILTER2	4
				DF_ZCINVDIFFOV	1
				DF_ZCINVVALUE	1
				DF_ZCNIO	18
				DF_ZCOBDEL2	6
				DF_ZCOBDELV	10
				DF_ZCOINCASES	13

Exemplary result

Monitoring – Failed Tasks 2/3

Find out which tasks failed the most in the system to troubleshoot and identify causes to avoid negative impact on the system due to e.g. outdated data.

Analytic Model: TCT_TASK_LOGS_AM_01

Exemplary Settings:
Filter on STATUS = FAILED
Rows:

- Trigger Type,
- Date → Week,
- Application ID

Column:

- Measure(Count)
- Date → Weekday

TCT_TASK_LOGS_AM_01

	Measures	ROW_COUNT							
	Weekday	(Null)	0_MON	1_TUE	2_WED	3_THU	4_FRI	5_SAT	6_SUN
TRIGGER_TYPE	Week								
Direct	(Null)	1	–	–	–	–	–	–	–
	15	–	1	8	13	1	2	–	–
	16	–	2	–	–	–	–	–	–
	17	–	11	1	2	–	5	–	–
	18	–	–	1	–	5	–	–	–
Scheduled	(Null)	53	–	–	–	–	–	–	–
	14	–	–	–	23	20	22	19	37
	15	–	28	26	81	26	10	11	44
	16	–	11	14	15	16	10	50	28
	17	–	11	15	14	15	17	19	24
	18	–	101	104	15	18	18	–	–

Exemplary result

Monitoring – Failed Tasks 3/3

Once it is identified what kind of tasks failed the most, you can check which error messages and the details of it.

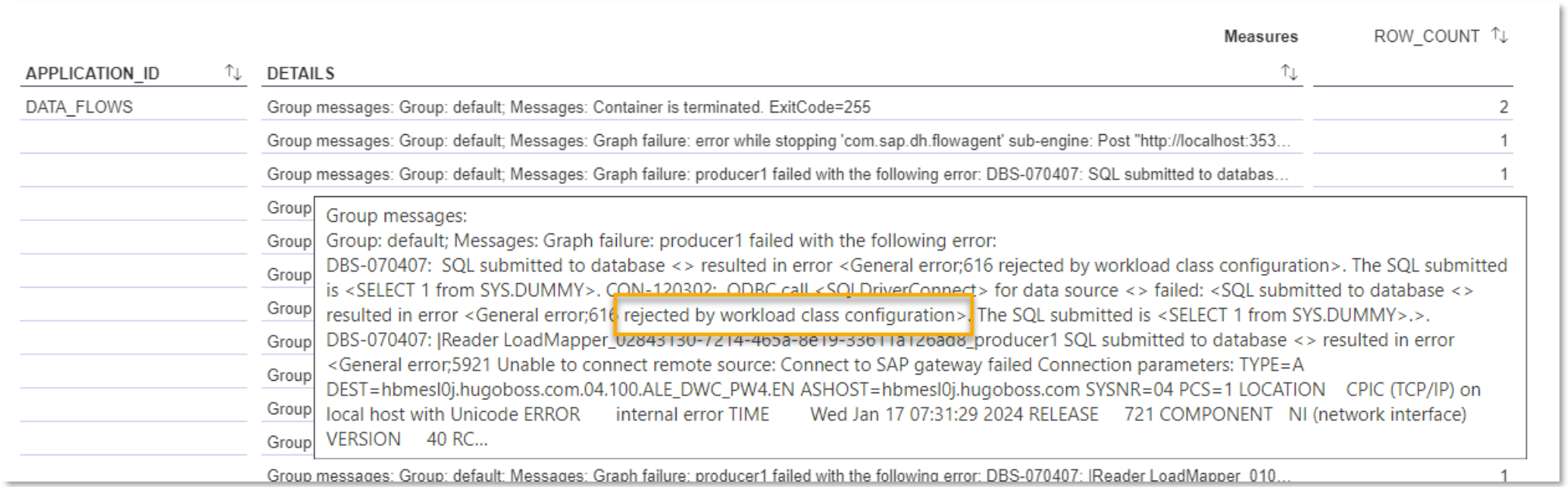
Analytic Model: TCT_TASK_LOG_MESSAGE_AM_01

Exemplary Settings:
Filter on SEVERITY = ERROR
Rows:

- Application ID
- Details

Column:

- Measure(Count)



Monitoring – Schedules

Check when replication processes are triggered by a schedule.

Analytic Model: TCT_TASK_LOGS_AM_01

Exemplary Settings:
Rows:

- Measure(Count)
- 5-minutes

Columns:

- Hours

Filter:

- Trigger Type: Scheduled
- Start Date: Single date

	Hours	0	1	2	4	5	6	7	8	9	10	12	13	14	16	18	20	22
START_DATE	Minutes-5																	
2024-02-12	0	–	–	7	2	1	12	16	14	1	16	18	4	7	7	7	6	7
	5	–	1	1	–	–	1	3	5	–	4	5	–	1	1	2	2	2
	10	–	–	–	1	–	2	–	–	–	2	4	–	2	2	1	2	1
	15	1	–	1	2	–	1	–	1	–	2	3	–	2	2	2	2	2
	20	–	–	1	–	1	–	–	–	–	–	–	–	–	–	–	–	–
	25	1	–	1	1	–	3	–	2	–	2	2	–	2	2	2	2	2
	30	–	–	–	–	1	4	2	1	–	1	1	–	1	1	1	1	1
	35	–	–	–	–	–	–	1	–	–	–	–	–	–	–	–	–	–

Exemplary result

Monitoring – User Activity

Check the user activity by e.g. getting an overview of the distinct users executing stories in SAC.

Analytic Model: TCT_MDS_STATEMENTS_STATISTICS_AM_01

Exemplary Settings:
Rows:

- Date → week

Columns:

- Measure(Distinct users),
- Date → Weekdays

Measures	Distinct User
Date-Day	
2024-02-05 (MON)	36
2024-02-06 (TUE)	53
2024-02-07 (WED)	55
2024-02-08 (THU)	49
2024-02-09 (FRI)	49
2024-02-10 (SAT)	2
2024-02-11 (SUN)	5
2024-02-12 (MON)	55
2024-02-13 (TUE)	58
2024-02-14 (WED)	55
2024-02-15 (THU)	65
2024-02-16 (FRI)	62
2024-02-17 (SAT)	3
2024-02-18 (SUN)	10
2024-02-19 (MON)	84
2024-02-20 (TUE)	83

Exemplary result

Measures	Distinct User		
Weekday	MON	TUE	WED
Week			
6	36	53	55
7	55	58	55
8	84	83	–

Exemplary result

Standard Monitoring

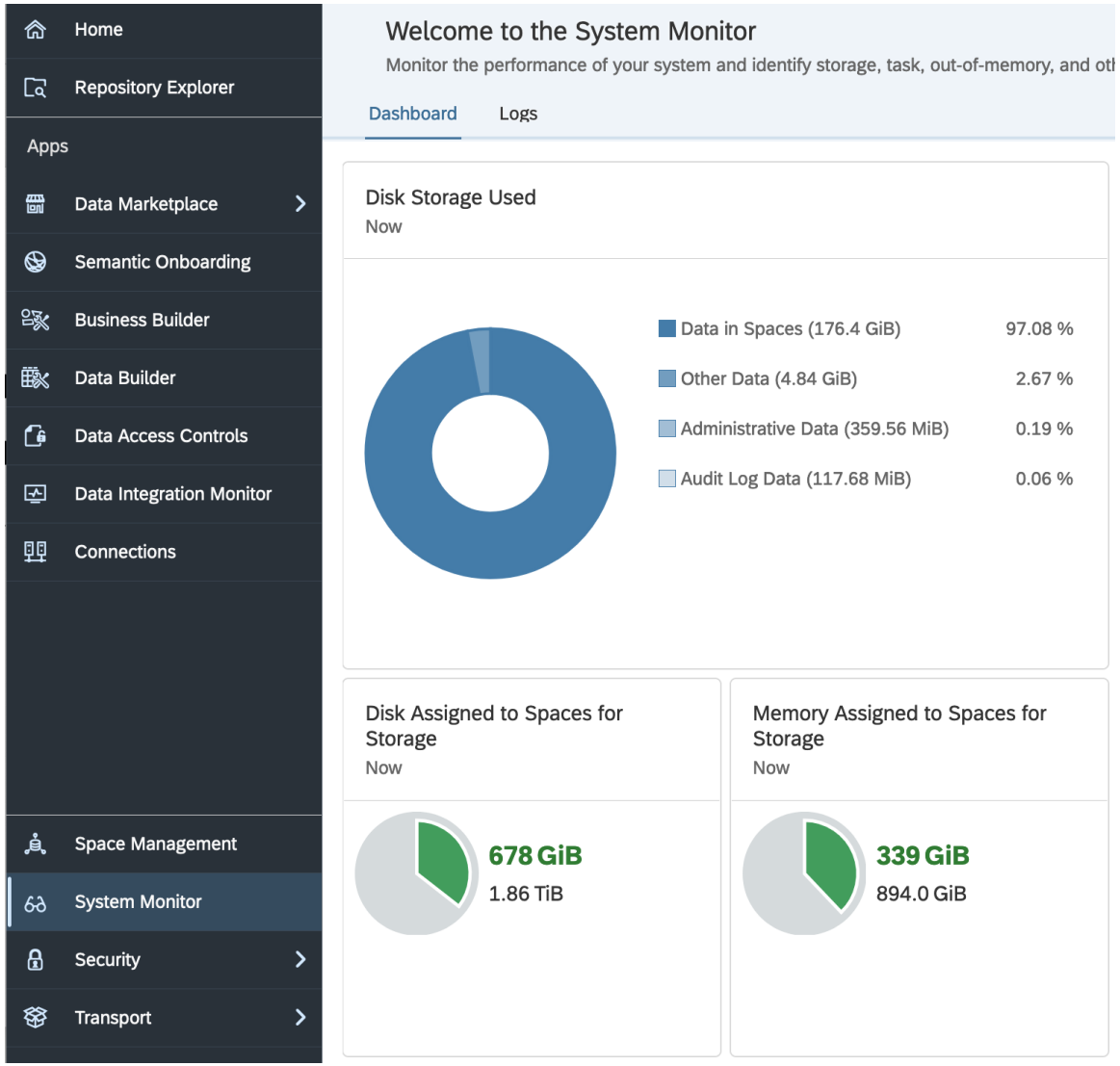
Monitoring in Datasphere Product: System Monitoring (Global)

Top 5 Ranking

- Out of Memory
- Long running MDS or task execution
- Admission Control
- Failed Tasks
- Relation to Spaces

Disk and Memory – Total Figures

Whole System in Scope

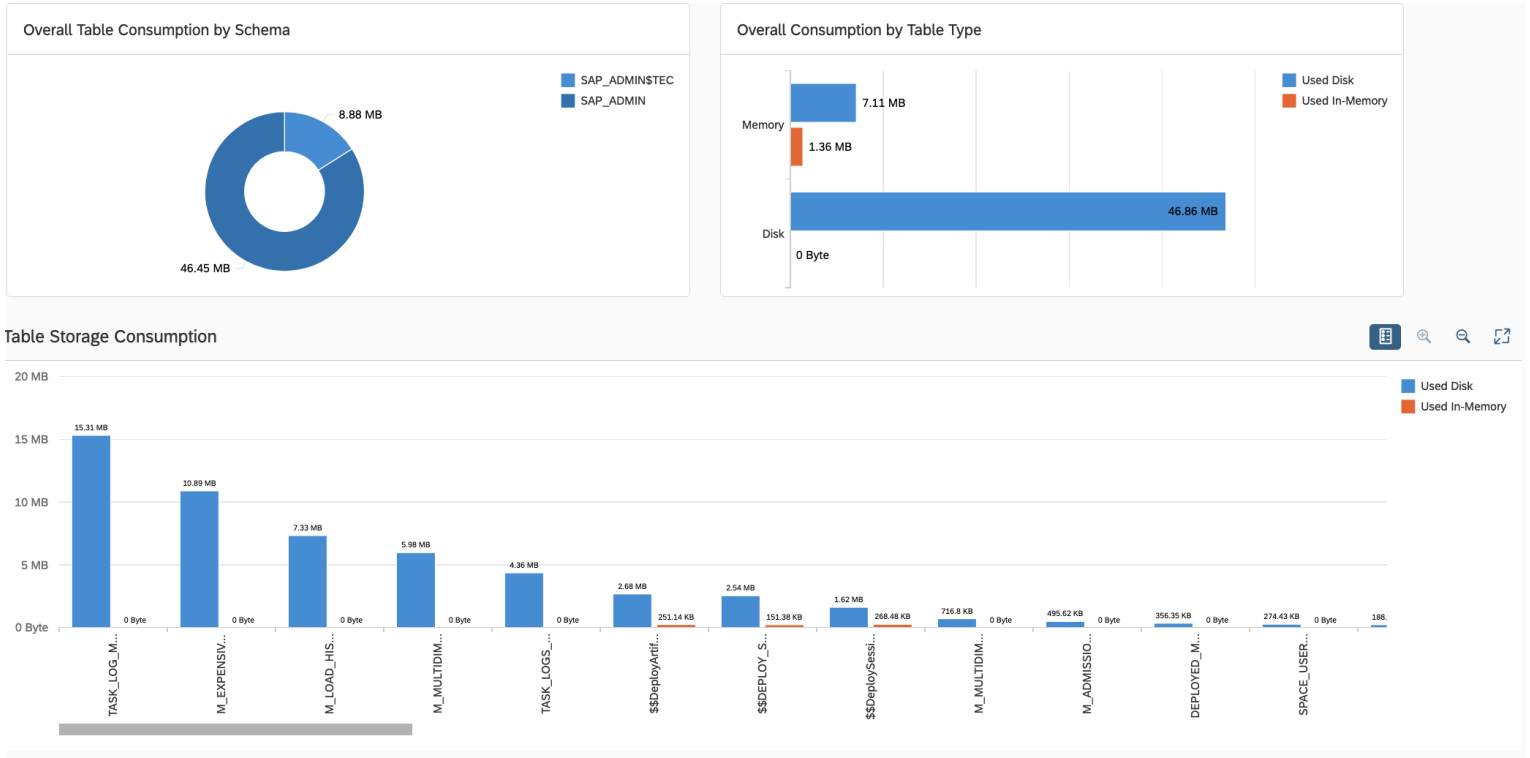


Monitoring in Datasphere Product: Space Monitor (per Space)

SAP Datasphere Space Monitor

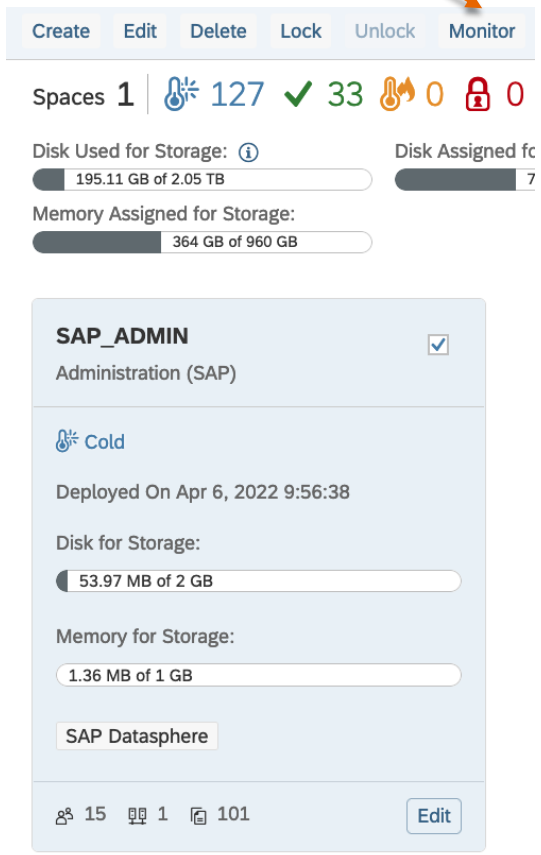
- List all tables and view persistencies of the space
- Memory on Disc and in Memory
- Record Count

This is available per Space.

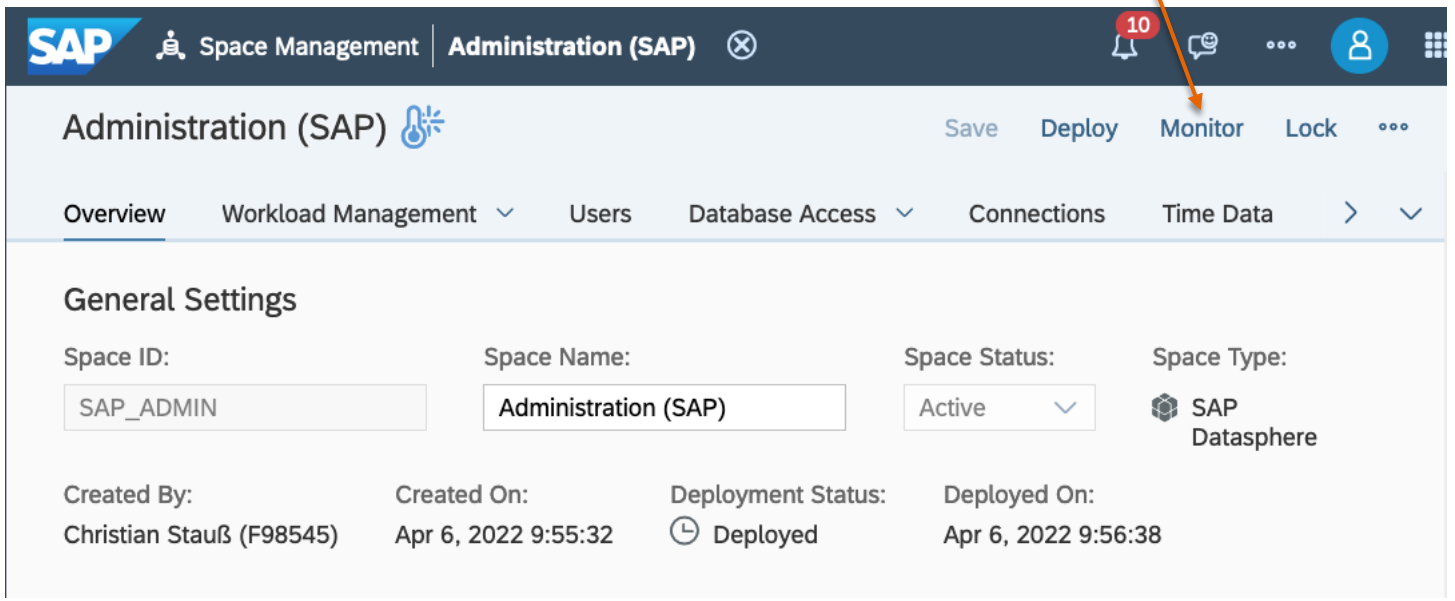


How to Navigate to the Space Monitor

Select your Space in the Space Management – and choose the “Monitor” button



Open the Space Details – and choose the “Monitor” button

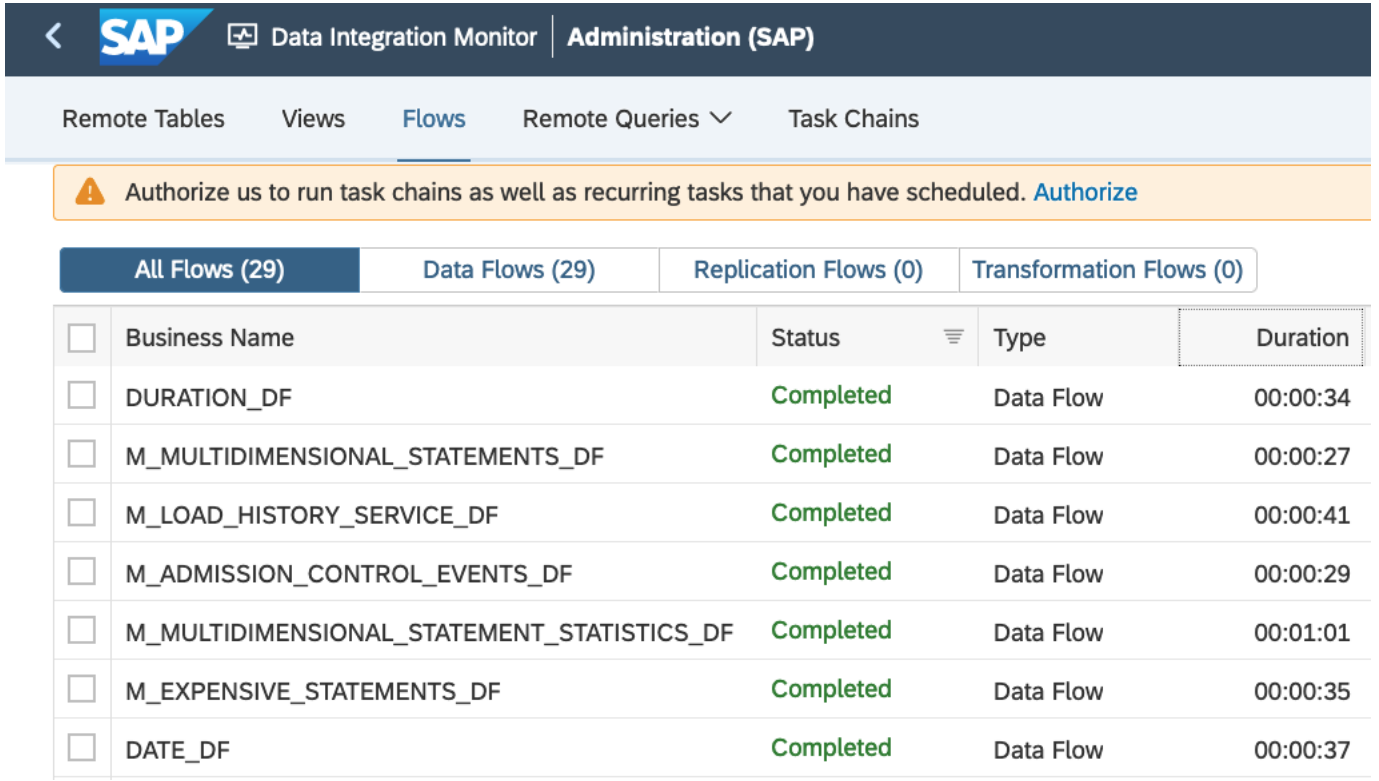


Monitoring in Datasphere Product: Data Integration Monitor (per Space)

Data Integration Monitor

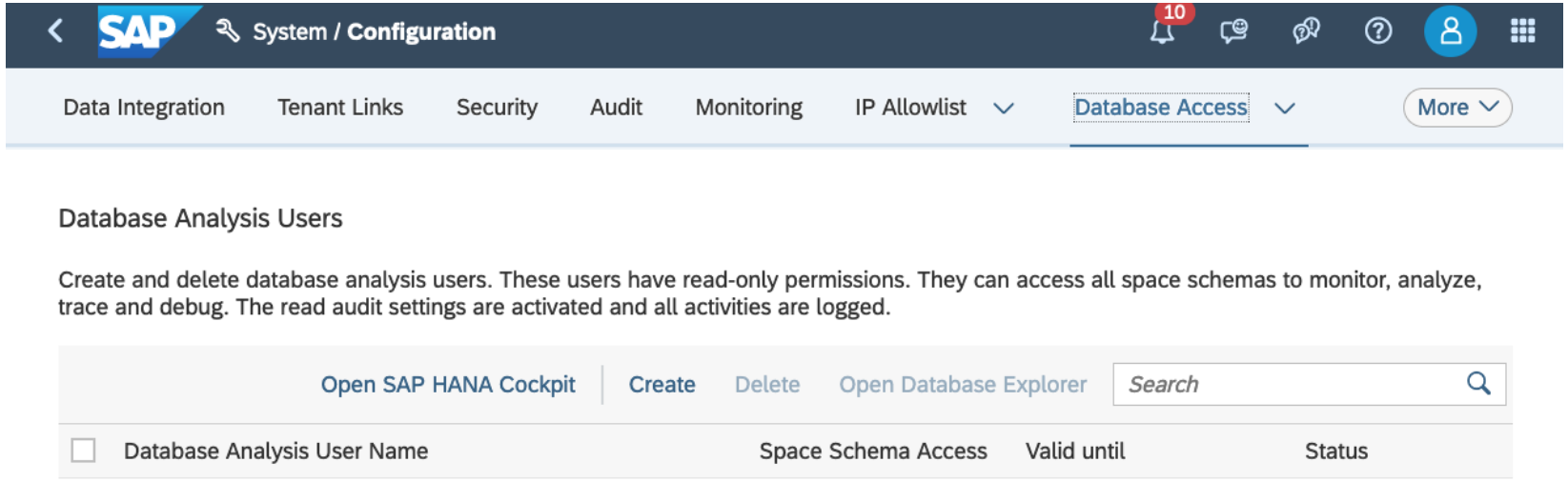
- Remote Table Status
- View Persistence
- Flows
- Remote Queries
- Task Chains

The portion of status, logs and runtimes is part of the monitoring content.



Monitoring in Datasphere Product: SAP HANA Cockpit (Global)

Launch the Cockpit from SAP Datasphere's Analysis User Section.

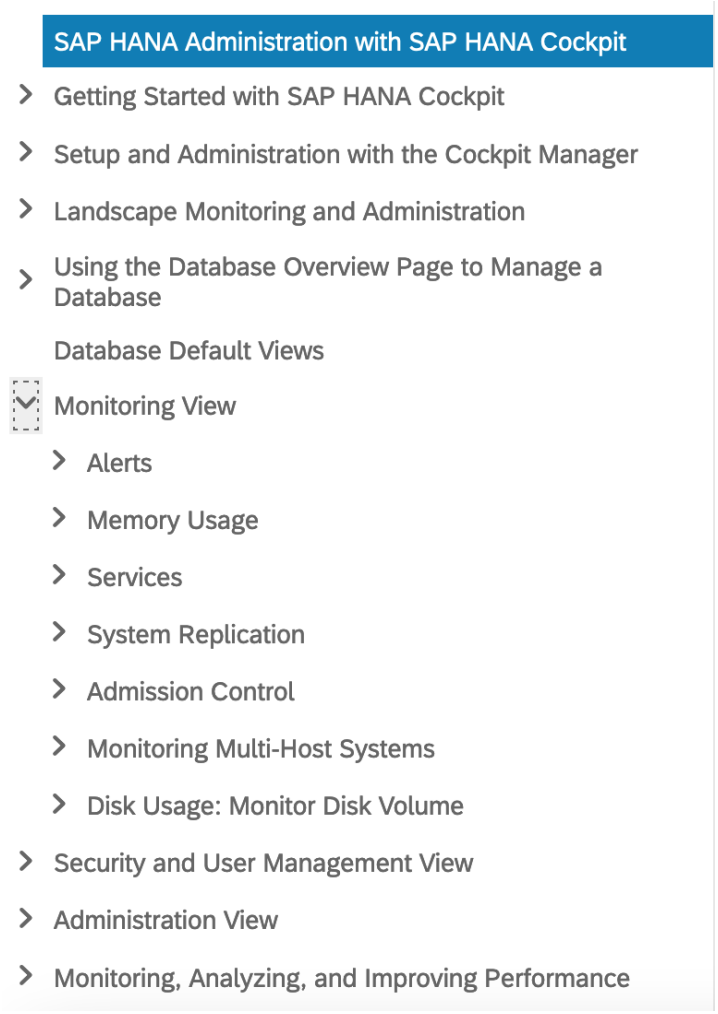


Monitoring in Datasphere Product: SAP HANA Cockpit

SAP HANA Cockpit is a native HANA Application to Monitor the various KPIs of SAP HANA Cloud.

See Documentation for more Details.

An Analysis User can be used to log into the SAP HANA Cockpit.



General Content Capabilites

Monitoring Content General Capabilities

Configurable Data Retention

- HANA Statistic Views typically capture a certain fixed amount of records only – and start overwriting the oldest entries.
- For monitoring, we need a reliable configurable retention time.
- Building up such a history is implemented for all relevant source tables.

Delta Replication

- For performance reasons, the update of the local persistence operates in a delta-mode.
- After an initial load, all subsequent data in loaded in deltas.

Realtime Data

- The reporting views can be implemented on views exposing replicated data or real-time data
- Missing delta in real-time mode is fetched during query runtime.
- This works similar to the Hybrid-Provider concept in BW.

Monitoring Content General Capabilties

Flexible Scope

- The delivered monitoring views have a scope per space or whole system.
- Monitoring offers various configurable granularities:
  - E.g. a set of task chains to a relevant operations team
  - E.g. a set of spaces (representing a project)
  - E.g. full system and even cross system

Authorizations and Access Control

- Administrative permissions are required to use the monitoring pages.
- This hinders the usage of delivered monitoring tools in daily work.
- The monitoring content comes with pre-defined data access controls allowing flexible assignment of data access according to the users role and task.

Monitoring Content General Capabilties

Custom UI Design

- Analytic Models in combination with SAP Analytics Cloud Reporting allow flexible UI design tailored to the target groups.
- Just build them on your own

Cross Tenant Scope

- When using more than one SAP Datasphere system, the data will be collected and stored in one place/tenant only.
- You can create cross system monitoring reports.
- All reports have to be build and deployed once.

Reporting:
Counting / Measure as Dimension

CPU as Measures **and** Dimension – Different Usage Patterns

Dimensions support
- Filtering & Sorting
- Description
- Attributes

Measures support
- Aggregation
- Filtering like Top10

Report: “Show Total CPU Time per Hour”
Here we like to aggregate the measure CPU Time per Hour

Report: “Show only records with a CPU Time above 60%”

We apply a filter on the dimension CPU Time.

This filter in independent from aggregation
– applied on a single record level

Scenario Example: CPU as Dimension - Counting

We like to count how many CPU Measurements are in a certain range:

- grouped in 10% steps
- three groups (green/yellow/red)
- ... and more ...

Analytical Model: TCT_HANA_COCKPIT_AM_01

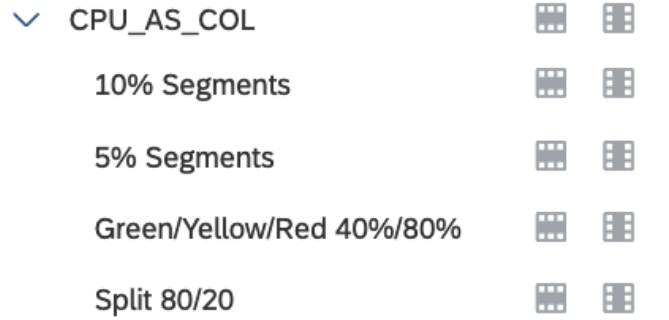
Rows:

- Attribute 10% Segments from Dimension CPU_AS_COL

Columns:

- Measure Count

	Measures	COUNT
10% Segments		
00% < 10%		66,459
10% < 20%		534
20% < 30%		83
30% < 40%		23
40% < 50%		10
50% < 60%		6
60% < 70%		1



Implementation: Define Attributes for Different Segments

Table PERCENTAGE_DIM

Percentage	10% Segments	5% Segments	Split 80/20	Green/Yellow/Red 40%/8(
0	00% < 10%	00% < 05%	< 80%	1_GREEN
1	00% < 10%	00% < 05%	< 80%	1_GREEN
2	00% < 10%	00% < 05%	< 80%	1_GREEN
3	00% < 10%	00% < 05%	< 80%	1_GREEN
4	00% < 10%	00% < 05%	< 80%	1_GREEN
5	00% < 10%	05% < 10%	< 80%	1_GREEN
6	00% < 10%	05% < 10%	< 80%	1_GREEN
7	00% < 10%	05% < 10%	< 80%	1_GREEN
8	00% < 10%	05% < 10%	< 80%	1_GREEN
9	00% < 10%	05% < 10%	< 80%	1_GREEN
10	10% < 20%	10% < 15%	< 80%	1_GREEN

The dimension table contains a mapping between the percentage value and segment it belongs to.

The columns show up as attributes in the reporting UI

Summary: “Show me the Distribution of …” involve counting of records.

Example:
How many Ina/MDS request to I receive per hour (segmentation by time)?

- Rows: Date, Hour (*Attribute of Time*)
- Columns: Measure Count

		Measures
Date-Day	Hours	EXECUTION_COUNT
2024-05-06 (MON)	2	9
	3	9
	6	862
	7	535
	8	227
	9	24
	11	708
	12	22

Example:
How many Ina/MDS request do I receive with a runtime larger than 30 secs (segmentation by runtime)?

- Rows: Date, Segment [10secs] (*Attribute of Runtime as Col*)
- Columns: Measure Count
- (Optional) Filter on Segment [10secs] > 30 secs

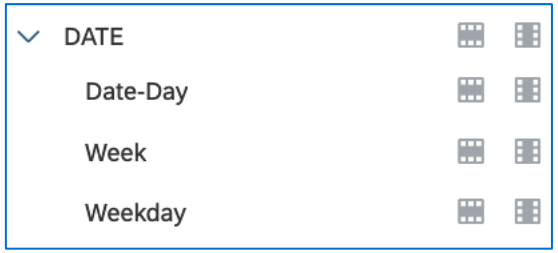
		Measures
Date-Day	Segment [10secs]	EXECUTION_COUNT
2024-05-06 (MON)	0 < 10	1,398
	00 < 10	987
	10 < 20	4
	20 < 30	2
	30 < 40	4
	40 < 50	1

Reporting –Dimensions Overview

Reporting Layer Specifics – Date Dimension

Date Dimension comes with the attributes

- Date-Day, e.g. 2024-05-06 (Mon)
- Week, e.g. 7
- Weekday, e.g. Mon, Tue, ...



Reporting Scenario: Compare Same Workdays

- Rows: Week
- Column: Weekday

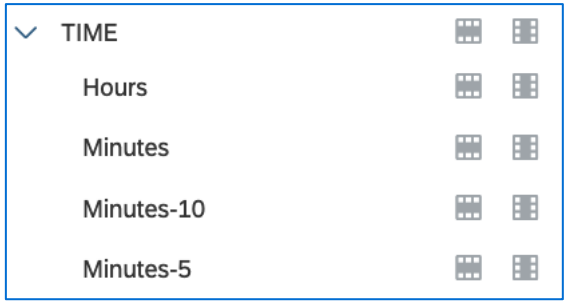
Similar Scenario:
Run evaluation only on Mon to Fri

Measures	EXECUTION_COUNT						
Weekday ⇄	Mon	Tue	Wed	Thu	Fri	Sat	Sun
Week ↑↓							
15	774	877	1,720	1,103	682	13	
16	478	755	669	1,281	1,106	28	
17	943	1,289	765	917	1,439		
18	637	758	33	236	410		258
19	2,420						

Reporting Layer Specifics – Time Dimension

Time Dimension derives from timestamp

- Hours
- Full Minutes (1, 2, 3, ...)
- Full Minutes-5 (5, 10, 15, ...)
- Full Minutes-10 (10, 20 , 30, ...)



Reporting Scenario: Time-Slices for Counting – e.g. request count per 5-minutes

- Rows: Date-Day, Hours, Minutes-5
- Column: Measure Count

			Measures	EXECUTION_COUNT
Date-Day	Hours	Minutes-5		
2024-05-06 (MON)	2	50		9
	3	10		9
	6	5		27
		15		202
		20		117
		25		44
		30		243
		45		128
		50		90
		55		11

Reporting Layer Specifics – Host Dimension

The host is not stable over time

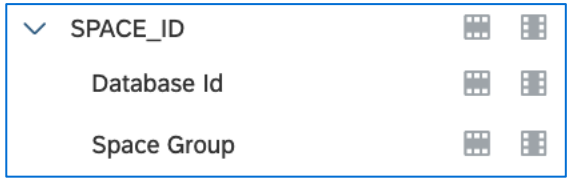
- Elastic Compute Node (ECN) usage adds/removes hosts
- A resizing of the system could lead to a new host name

The Host-Dimension maps all hosts to a stable database Id (as displayed in the about-dialog).

It is recommended to use the database id in reporting.

Reporting Layer Specifics – Space Dimension

The space has the attribute „Space Group“.



Example:

- ABC_SPACE1
- ABC_SPACE2

➔ Both belong to group ABC.

Rule: The text before the first underscore is used as group.
The rule is implemented in view SPACE_SRC.

Reporting Layer Specifics – SAC Story/Widget Dimension

The monitoring content extract the UIDs of a Story and Widget only.

Story attributes like the description can be downloaded from SAP Analytics Cloud imported into SAP Datasphere.

The same is possible for Widget attributes like descriptions and type.

Procedure for Download:

- Logon to your SAP Analytics Cloud tenant
- Navigate model SAC_PERFORMANCE_E2E
- Open the model in the Data Analyzer
- Select the relevant columns and download the result to a CSV file.
- Upload the files into the SAP Datasphere tables SAC_PERFORMANCE_E2E_REMOTE_WIDGET respective SAC_PERFORMANCE_E2E_REMOTE_STORY

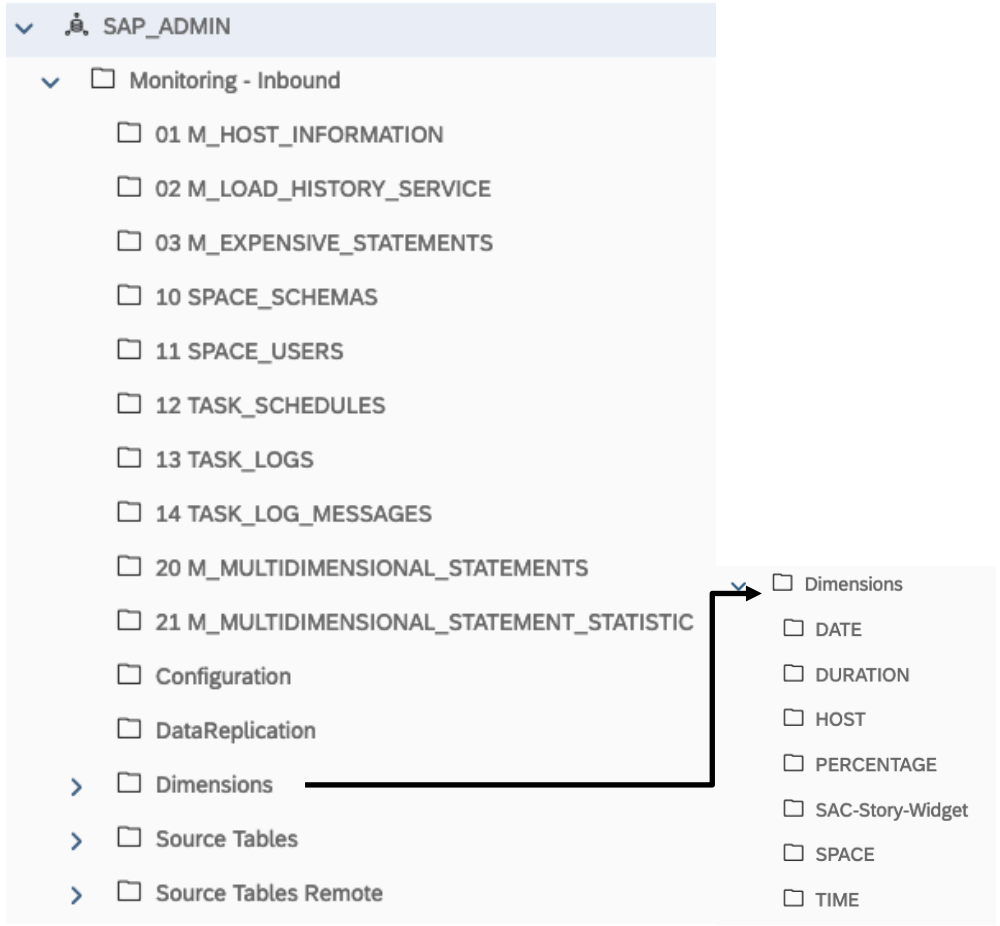
Column Names

Business Name	Technical Name
STORY_ID	STORY_ID
STORY_TXT	STORY_TXT

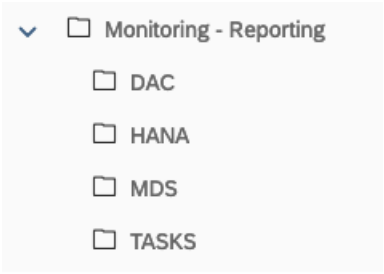
Business Name	Technical Name
WIDGET_IDS	WIDGET_IDS
WIDGET_TXT	WIDGET_TXT
WIDGET_TYPE	WIDGET_TYPE

Monitoring Content Overview

Data Inbound (Views, Dataflows, ...)



Reporting Views and Data Access Control



Configuration Tables

Monitoring Data – Table Growth

The numbers of records added per day typically depends on your system activities.

Below list shows the expected memory on disc per records.

Based on your real number of new records per day, calculate the disc space required for e.g. 30 days.

Table Name	Disk Usage per x records	Records per Day
M_EXPENSIVE_STATEMENTS	17,6 MB / 100.000 recs	
M_MULTIDIMENSIONAL_STATEMENTS	10,0 MB / 100.000 recs	
M_MULTIDIMENSIONAL_STATEMENT_STATISTICS	14,1 MB / 100.000 recs	
M_LOAD_HISTORY_SERVICE	4,4 MB / 100.000 recs	8640 x number of Ports (2 to 3) => approx. 25.000 records
M_HOST_INFORMATION	0.2 MB / 100 recs	no relevant increase
SPACE_SCHEMAS	0,1MB / 1000recs	no relevant increase
SPACE_USERS		no relevant increase
TASK_LOGS_V_EXT	5,1 MB / 100.000 recs	
TASK_SCHEDULES		no relevant increase
TASK_LOG_MESSAGES_V_EXT	4,3 MB / 100.000 recs	

Reporting - Authorizations

Introduction Data Access Controls (DAC)

The content contains three Data Access Controls:

- TCT_DAC_SPACE_ID_DAC
  - List of spaces a user is authorized for
  - Maintained in table TCT_DAC_APP_USER_V_01

- TCT_DAC_SPACE_DATABASE_DAC
  - Almost the same as above
  - Accepts a combination of Space and Database Id
  - If data contains data of two tenants, the space is only unique in combination with the database id.

- TCT_DAC_APP_USER_DAC
  - Dedicated for HANA views containing an application user column
  - Return per default the current user: everybody can see his data in HANA tables

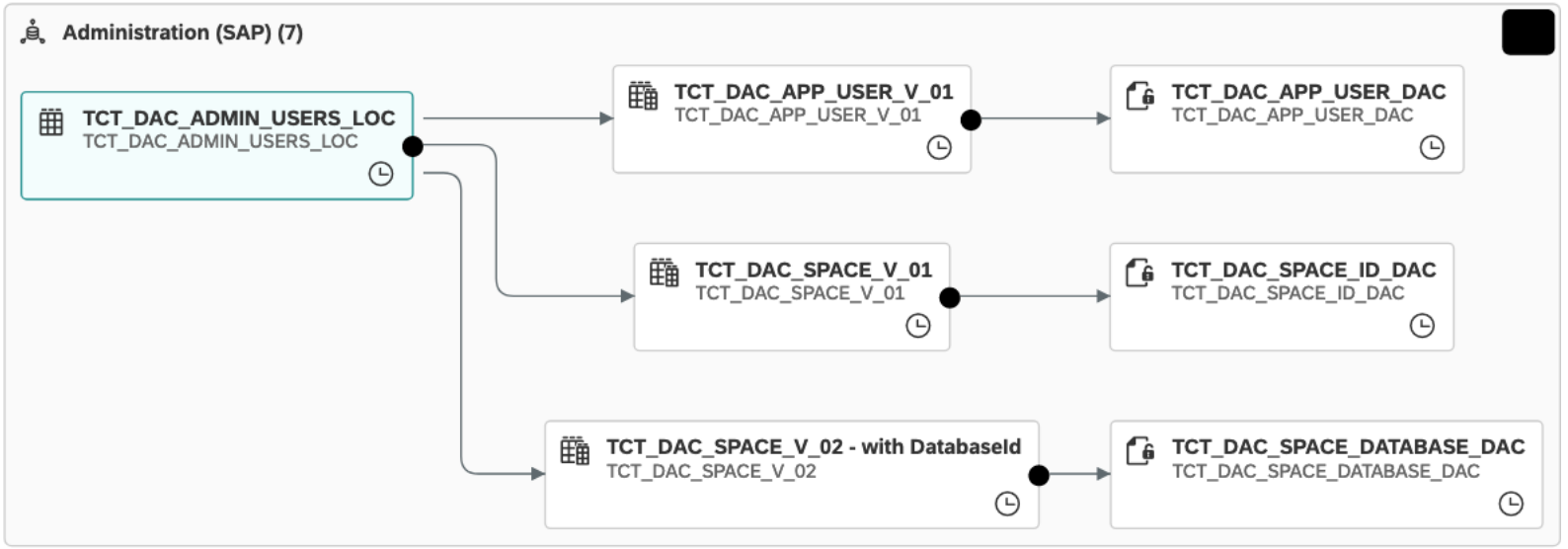
Introduction Data Access Controls – Admin Access

The table TCT_DAC_ADMIN_USERS_LOC lists all administrators.

Users listed in this table have full access to all data.

Users can be enabled/disabled – no need to delete and re-enter the name.

Business Name	Technical Name
Admin User	APP_USER
Enabled	ENABLED



Using Data Access Controls

The general recommended approach is to assign the DAC at the view closest to the Analytical Model. As the DAC released in Q1-2024 add filters (instead of inner joins) the values are propagated (“unfolding of selections”).

Step-by-Step:

- Open the Fact-view feeding the Analytical Model you like to protect.
- Navigate to the Data Access Control in the Model Properties.
- Select the Data Access Control – e.g. TCT_DAC_APP_USER_DAC for a HANA view.
- Map the view column containing the application user with column APP_USER
- Save and Deploy.

Modelling Session – Q&A

Links to Blog Posts

SAP Datasphere: HANA System Memory and CPU - Overall Consumption and Breakdown

SAP Datasphere: Monitoring and Analysis of SAC requests (Ina/MDS)

HANA Analytical Privileges in SAP DWC - Part 1

SAP Data Warehouse Cloud: Data Integration Monitoring – Running Task Overview

SAP Data Warehouse Cloud: Data Integration Monitoring - Sample Content for Reporting

Thank you.

Contact information:

Olaf Fischer

olaf.fischer@sap.com

LinkedIn